Appl. No. 09/599,812

AMENDMENTS TO THE CLAIMS

Please amend the claims of the present application as set forth in the listing of claims below. This listing of claims will replace all prior versions, and listings, of claims in the application.

Claims 1-42 were originally filed.

Claims 1-26, 33, and 40 have been amended.

New claims 43-45 have been added.

No claims have been cancelled.

Accordingly, claims 1 – 45 are pending.

LISTING OF CLAIMS

1.   (Currently Amended) A <u>computer-readable medium embodying a</u> software architecture ~~embodied on a computer-readable medium~~, the architecture comprising:

multiple attachment points collectively arranged to filter data associated with files that describe software extensions<u>, each attachment point including logic for filtering data</u>; and

multiple extension managers associated with the multiple attachment points and with respective feature types that can be added to a software platform by software extensions, the extension managers being configured to receive data from the multiple attachment points that pertains only to the feature type with which the extension manager is associated.

Appl. No. 09/599,812

2. (Currently Amended) The ~~software architecture~~ <u>computer-readable medium</u> of claim 1, wherein the attachment points are defined as predicate chains.

3. (Currently Amended) The ~~software architecture~~ <u>computer-readable medium</u> of claim 1, wherein the attachment points filter XML data.

4. (Currently Amended) The ~~software architecture~~ <u>computer-readable medium</u> of claim 3, wherein each feature type is associated with an XML tag.

5. (Currently Amended)v The ~~software architecture~~ <u>computer-readable medium</u> of claim 3, wherein each feature type is associated with an XML tag, at least some of the feature types comprising user-defined feature types.

6. (Currently Amended) The ~~software architecture~~ <u>computer-readable medium</u> of claim 1, wherein each attachment point exposes collections of ordered nodes.

7. (Currently Amended) The ~~software architecture~~ <u>computer-readable medium</u> of claim 1, wherein each attachment point exposes collections of ordered XML nodes.

LEE & HAYES, PLLC 4 MSI-586US.M03

PAGE 6/30 * RCVD AT 5/3/2004 2:09:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/2 * DNIS:8729306 * CSID:3035390271 * DURATION (mm-ss):07-06

Appl. No. 09/599,812

8. (Currently Amended) A computer <u>system including the computer-readable medium of claim 1 and implementing</u> ~~embodying~~ the software architecture ~~of claim 1~~.

9. (Currently Amended) A <u>computer-readable medium embodying a</u> software architecture ~~embodied on a computer-readable medium~~, the architecture comprising:

a hub structure configured to:

receive multiple different files that describe extensions that can be added to a software platform;

combine the multiple different files into a single exposable list; and

expose the single exposable list to a filter structure comprising one or more attachment points configured to filter the list<u>, each attachment point including logic for filtering</u>.

10. (Currently Amended) The ~~software architecture~~ <u>computer-readable medium</u> of claim 9, wherein the hub structure receives multiple different XML files and exposes a list of XML nodes.

11.   (Currently Amended)   A computer system including the computer-readable medium of claim 9 and implementing ~~embodying~~ the software architecture ~~of claim 1~~.

12.   (Currently Amended) A computer-readable medium embodying a software architecture ~~embodied on a computer-readable medium~~, the architecture comprising multiple different attachment points each of which ~~is configured~~ includes logic making the attachment point operable to:

receive XML data that pertains to one or more software extensions that can be added to a software platform;

process the XML data to provide a list of XML nodes; and

expose the list of XML nodes.

13.   (Currently Amended)   The ~~software architecture~~ computer-readable medium of claim 12, wherein the list of XML nodes is exposed to another attachment point.

14.   (Currently Amended)   The ~~software architecture~~ computer-readable medium of claim 12, wherein the list of XML nodes can pertain to multiple different feature types that can be added by the one or more software extensions.

LEE & HAYES, PLLC   6   MSI-586US.M03

PAGE 8/30 * RCVD AT 5/3/2004 2:09:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/2 * DNIS:8729306 * CSID:3035390271 * DURATION (mm-ss):07-06

Appl. No. 09/599,812

15.   (Currently Amended)   The ~~software architecture~~ <u>computer-readable medium</u> of claim 12, wherein the list of XML nodes can pertain to multiple different features of particular feature types that can be added by the one or more software extensions.

16.   (Currently Amended)   The ~~software architecture~~ <u>computer-readable medium</u> of claim 12, wherein the list of XML nodes can pertain to one or more of:

multiple different feature types that can be added by the one or more software extensions; and

multiple different features of particular feature types that can be added by the one or more software extensions.

17.   (Currently Amended)   A computer <u>system including the computer-readable medium of claim 12 and implementing</u> ~~embodying~~ the software architecture ~~of claim 1~~.

18.   (Currently Amended) A <u>computer-readable medium embodying a</u> software architecture ~~embodied on a computer-readable medium~~, the architecture comprising:

a hub structure configured to:

Appl. No. 09/599,812

receive multiple different files that describe software extensions that can be added to a software platform;

combine the multiple different files into a single exposable list; and

expose the single exposable list to a filter structure that is configured to filter the list,

the filter structure comprising multiple attachment points collectively arranged to filter data associated with the list exposed by the hub structure, each attachment point including logic for filtering data; and

multiple extension managers associated with the multiple attachment points and with respective feature types that can be added to a software platform by software extensions, the extension managers being configured to receive data from the multiple attachment points that pertains only to the feature type with which the extension manager is associated.

19.   (Currently Amended)   The ~~software architecture~~ computer-readable medium of claim 18, wherein the hub structure receives multiple different XML files and exposes a list of XML nodes.

20.   (Currently Amended)   The ~~software architecture~~ computer-readable medium of claim 19, wherein the list contains root node tags for all of the XML files.

LEE & HAYES, PLLC   8   MSI-3X6US.M03

PAGE 10/30 * RCVD AT 5/3/2004 2:09:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/2 * DNIS:8729306 * CSID:3035390271 * DURATION (mm-ss):07-06

21.   (Currently Amended)   The ~~software architecture~~ <u>computer-readable medium</u> of claim 19, wherein the XML files logically describe where a particular extension fits on the software platform.

22.   (Currently Amended)   The ~~software architecture~~ <u>computer-readable medium</u> of claim 19, wherein the attachment points are defined as predicate chains.

23.   (Currently Amended) The ~~software architecture~~ <u>computer-readable medium</u> of claim 19, wherein an extension manager is notified whenever an extension comprising a feature type with which it is associated is added or removed from the software platform.

24.   (Currently Amended)   The ~~software architecture~~ <u>computer-readable medium</u> of claim 19, wherein each feature type is associated with a particular XML tag.

25.   (Currently Amended)   A computer <u>system including the computer-readable medium of claim 18 and implementing</u> ~~embodying~~ the software architecture ~~of claim 1~~.

26.   (Currently Amended)   A   method   of   providing   a   software

extension comprising:

exposing an XML list that contains one or more nodes;

processing the XML list <u>with a structure including multiple attachment</u>

<u>points</u> to identify specific nodes that correspond to various feature types that can

be added to a software platform, <u>at least one of the attachment points including</u>

<u>processing logic</u>; and

notifying an extension manager that is associated with at least one feature

type if a node that corresponds to that feature type is identified in the XML list.

27.   (Original)   The method of claim 26, wherein said processing is

accomplished by filtering the XML list using multiple attachment points that are

defined as predicate chains.

28.   (Original)   The method of claim 27, wherein the individual

attachment points receive XML data as an input and expose a list of XML nodes.

29.   (Original)   The method of claim 26, wherein said processing is

accomplished by filtering on specific nodes.

30.   (Previously Presented) The method of claim 26, wherein said

processing is accomplished by exposing various nodes.

LEE & HAYES, PLLC   10   MSI-586US.MOS

PAGE 12/30 * RCVD AT 5/3/2004 2:09:06 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-1/2 * DNIS:8729306 * CSID:3035390271 * DURATION (mm-ss):07-06

31.   (Previously Presented) The method of claim 26, wherein said processing is accomplished by filtering on specific nodes and exposing various nodes.

32.   (Original)   One or more computer-readable media having computer-readable instructions thereon which, when executed by a computer, cause the computer to implement the method of claim 26.

33.   (Currently Amended)   A method of providing a software extension comprising:

receiving XML data that pertains to a software extension that is to be added to a software platform;

processing the XML data using one or more attachment points to identify XML nodes, at least one of the attachment points including logic; and

exposing an XML list that contains one or more nodes that are identified by said processing.

34.   (Original)   The method of claim 33, wherein said receiving comprises receiving multiple XML files that pertain to different software extensions.

35.   (Original)   The method of claim 34, wherein said processing comprises combining the multiple XML files into a single exposable list.

36.   (Previously Presented)   The method of claim 33, wherein the XML list is exposed to one or more attachment points that are defined as predicate chains that filter the XML data.

37.   (Previously Presented) The method of claim 36, wherein at least one of the attachment points exposes a node.

38.   (Original)   The method of claim 36, wherein at least one of the attachment points filters on a node.

39.   (Original)   One or more computer-readable media having computer-readable instructions thereon which, when executed by a computer, cause the computer to implement the method of claim 33.

40.   (Currently Amended) A method of providing a software extension comprising:

receiving multiple different files, each of which being associated with a different software extension and logically describing its associated software extension;

combining the multiple different files in a single list;

exposing portions of the list;

processing exposed portions of the list <u>using one or more attachment points</u> to identify one or more feature types that are to be added to a software platform, <u>at least one of the attachment points performing a logical operation</u>; and

notifying an extension manager that is associated with a particular feature type.

41.   (Original)   The method of claim 40, wherein the multiple different files comprise XML files.

42.   (Original)   One or more computer-readable media having computer-readable instructions thereon which, when executed by a computer, cause the computer to implement the method of claim 40.

43.   (New)   A computer-readable medium embodying a software architecture, the architecture comprising:

a collection of attachment points, each attachment point including logic that allows the attachment point to filter data from a list of software extension description files; and

one or more extension managers, each extension manager being associated with a feature types that can be added to a software platform by a software

extension, each extension manager being configured to receive only data from the list of software extension description files that pertains to the feature type associated with the extension manager.

44.   (New)   A computer-readable medium embodying a software architecture, the architecture comprising:

a hub structure configured to:

combine into a single exposable list multiple different files, each file describing an extension that can be added to a software platform; and

filter the single exposable list using a filter structure comprised of multiple objects, each object including logic that allows the object to filter data from the list.

45.   (New)   A method of providing a software extension comprising:

processing an XML list including one or more nodes using a collection of attachment points, each attachment point including logic that allows the attachment point to identify a specific node that correspond to various feature types that can be added to a software platform; and

notifying an extension manager that is associated with at least one feature type if a node that corresponds to that feature type is identified in the XML list.

14